SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V13I9P106

Volume 13 Issue 9, 83-100, September 2026

Original Article

HyperFusionNet: An Intelligent DRL Framework with Neural Attention and Memory Integration for Optimized IoT Communication and Energy Efficiency

V. Satya Sudha[1], Syed Shameem[2*]

[1,2]*Department of Electronics and Communication Engineering, Koneru Lakshmaiah Education Foundation, K L Deemed to be University, Vaddeswaram, AP, India.*

[2*]*Corresponding Author : shameemsyed324@gmail.com*

Received: 29 April 2026 Revised: 07 August 2026 Accepted: 19 August 2026 Published: 29 September 2026

Abstract - *The sheer increase in IoT networks has created major challenges in the effective management of data communication, and high latency, limited power supply resources, and unreliable networks are some of the problems that have been encountered. All these limitations pose challenges to the scalability and performance of the IoT system, and more sophisticated data routing, energy consumption, and network stability optimization solutions are required. In this study, it is suggested to use HyperFusionNet, a progressive Deep Reinforcement Learning (DRL) model aimed at maximizing communication in a large-scale IoT network. HyperFusionNet extends the use of the attention and memory modules in neurons with the objective of reducing latency, improving throughput and increasing network lifetime, i.e., tackling the limitations of energy and large loads of data in IoTs. The framework was evaluated over 100 simulated episodes and converged within 80 episodes, achieving a normalized average cumulative reward of 960. Across the evaluated conditions, HyperFusionNet maintained latency of 90-120 ms, a throughput of 900-1200 bps, PDR of 95-99%, and an average network power of 1000-1050 mW. Under the common experimental pipeline, it provided approximately 15% lower latency, 12% higher throughput, and a 20% improvement in network lifetime relative to the internal reference configuration. Also, the energy used per node was reduced by almost one-fifth that of other routing protocols, increasing life span of the network by nearly one-fifth. The DRL agent used in HyperFusionNet is the Proximal Policy Optimization (PPO) that has curiosity-based exploration and reward shaping strategy to adapt to real-time network conditions. The model converged in 80 episodes with an average cumulative reward of 960 and a robustness of their policy that retained 95% reliability in the unfavorable conditions, e.g., higher traffic and node failures.*

Keywords - *Deep Reinforcement Learning, HyperFusionNet, Internet of Things, Packet delivery ration, Proximal Policy Optimization.*

1. Introduction

Internet of Things (IoT) is a technological revolution which allows billions of devices all over the world to interact, share information, and communicate with each other. These IoT networks have a variety of tasks, such as smart homes, industrial automation, healthcare, environmental monitoring, and agriculture. All these applications have their specific needs, including real-time data exchange requirements, low latency, high reliability, and, most importantly, energy efficiency [1]. To ensure effective communication in the IoT settings, it is necessary to use protocols that are capable of addressing these needs and supporting the scale and limitations of the IoT devices, including limited bandwidth, power, and processing resources [2]. The IoT communication protocols may be classified into short-range communication protocols and long-range transmission protocols, each of which has its application based on their transmission range, data rate, and energy consumption. Bluetooth Low Energy (BLE) finds broad application in applications where low power usage and short-range communications are important, like in a fitness tracker, wearable device, and smart home applications. BLE utilities 2.4 GHz ISM band, and it is optimized to deliver small data volume with minimal energy usage and a range of about 100 meters [3]. Zigbee that boasts of energy efficiency and mesh networking capabilities, Zigbee is commonly applied in smart-home and factory applications. It works within the frequency of 2.4 GHz and can provide the ability of transmitting data to several devices in a network, which expands the range of the network and yet saves power [4]. Wi-Fi is also common among high-bandwidth applications that need high data rates, e.g. security cameras and smart televisions. Wi-Fi has a larger power consumption than other protocols such as BLE and Zigbee, which restrict its application in battery-powered IoT devices. The Wi-Fi has a range of approximately 100-200 meters and consumes large power to maintain connectivity [5]. LoRaWAN, Long Range Wide Area Network (LoRaWAN) becomes widely used protocol in IoT systems that need to be long range, low power, including smart agriculture and environmental monitoring. LoRaWAN has unlicensed frequency bands, which means it can transmit data across several kilometers, and therefore is the best option when deploying to remote locations where constant replacement of batteries is unfeasible. Narrow band IoT belongs to the family of cellular technology, which is made to work on licensed cellular networks and can ensure reliable long-range communication [6, 7].

The rise of IoT networks around the world makes energy consumption a more significant issue. IoT applications are often used in inaccessible or remote areas and powered by batteries, and energy efficiency is a critical consideration. The powering of the IoT devices will guarantee the uninterrupted and stable functioning of the equipment, decrease the expenses of upkeep, and decrease the environmental footprint [8, 9]. As networks of IoT grow, so do energy requirements. An example of this is smart city implementation that may involve thousands of interlinked sensors, which need power to operate and collect, process and transmit data. Scalability requires energy efficiency in all devices since inefficient operation efficiency can quickly exhaust batteries, thus constraining the operational cycle of the system.

Although there is progress in IoT technology, there are still a few challenges in the way of optimizing communication performance in terms of energy efficiency, reliability, scalability and security. Such issues need address to ensure the full potential of the IoT networks is realized and to support a broad set of applications [10]. The low-latency of communication is needed in applications that require real-time data exchange and decisions, including autonomous vehicles, industrial automation, and telemedicine. Other protocols, such as LoRaWAN, are energy efficient, however they might not support the latency criteria of these applications because of multi-hop routing, network congestion or interference in shared frequency bands.

The edge computing is used to process the data near a data source, and as a result, the data being transmitted to the cloud is minimized, and consequently, the latency is minimized [11]. Multi-path routing algorithms, which spread out network traffic along more than one path, are used to cope with congestion in the network, enhancing data delivery speed and latency. The credibility of data rate in the IoT networks is of paramount importance, particularly in the applications in healthcare, industrial control and smart infrastructure. This can be enhanced by the implementation of error correction schemes, adaptive modulation schemes and dynamic power adjustment schemes that enhance reliability in the transmission. Moreover, the PDR can be improved by the use of protocols that choose the best transmission routes, according to real-time conditions, without consuming too much energy [12]. Now that there is a lot of data that is being exchanged in the IoT networks, data security is of utmost importance. The IoT devices are prone to cyberattacks, including data interception, unauthorized access, and Denial of Service (DoS) attacks, since they do not have the resources to provide strong security protocols. Also, a significant proportion of IoT devices communicate through public networks, which puts the issue of data breaches and lapses in data security increases [13].

Deep Reinforcement Learning (DRL) techniques achieves the potential to optimize routing with real-time learning and adaptation to network conditions, minimizing energy use, and lowering latency and reliability in IoT networks. Fog and edge computing bring the data processing as close to the data source as possible, reducing thereof latency, bandwidth consumption, and allowing real-time analytics in critical applications such as autonomous systems and healthcare. Software-Defined Networking (SDN) enables control of traffic at its center, which enables dynamic routing control, traffic prioritization and resource optimization in order to achieve energy efficiency.

The IoT devices present significant challenges, especially in communication and network management. IoT networks are also defined by a wide range of devices with different capabilities in terms of computational power, connectivity, and energy limitation, which work in a dynamic environment, with network conditions changing quickly. These attributes cause an urgent demand of effective communication protocols that are capable of supporting high data loads, supporting low latency, and saving on the limited battery power on devices all without compromising data delivery in a reliable manner. The HyperFusionNet design meets these issues by appealing to the latest Deep Reinforcement Learning (DRL) models by adding neural attention and memory units to enable a more efficient, responsive, and sustainable IoT communication.

IoT communication protocols and learning-based models enhanced the routing, resource allocation, and energy management. However, the existing solutions have many limitations. Traditional routing and scheduling methods are static, rule-based, and protocol-dependent. This makes them ineffective under dynamic traffic, unstable link, packet loss, and node-energy variations. Many DRL-based IoT models optimize individual objectives such as latency, energy, or reliability. They do not balance latency, throughput, packet delivery ratio, and energy consumption together.

To solve such problems, proposed HyperFusionNet is a better candidate. It is an intelligent DRL-based framework designed to optimize IoT communication. It models the proposed IoT communication control the sequential decision-taking phenomenon, where the agent observes the current network state, and uses neural attention to prioritize critical features. It then applies an LSTM-based memory module to retain temporal network behavior. From this, it learns adaptive routing, priority management, and wake-sleep scheduling policies through proximal policy optimization. The main elements of this model are

- To develop an adaptive DRL-based IoT communication framework which selects optimal routing paths in real-time communication, such as latency, congestion, packet loss, and link reliability.
- To integrate neural attention and LSTM-based memory mechanisms for prioritizing critical network-state features. This helps in preserving historical traffic behavior and enables informed and stable decision-making in dynamic IoT environments.
- To design a multi-objective reward function which optimizes latency, throughput, energy consumption, and packet delivery ratio together. This helps in balancing communication efficiency, reliability, and network lifetime.
- To improve the robustness and adaptability of IoT communication under challenging conditions such as high traffic load, node failure, and energy-constrained operation, using PPO-based stable policy learning and adaptive wake-sleep scheduling.

The uniqueness of the proposed HyperFusionNet model is present in its integrated attention-memory-policy optimization structure for multi-objective IoT communication. Conventional models, such as attention-based, memory-based, energy-aware, or PPO-based DRL methods, address these functions separately. HyperFusionNet integrates them into a single adaptive framework. This allows the model to prioritize critical network features and preserve historical traffic patterns. This helps the model to learn stable routing and scheduling policies and optimize complete system parameters.

In order to train the DRA frame to choose trustworthy routes, it is crucial not to consider regions with a high packet loss and interference. Through the dynamically executed routing strategies, which adapt dynamically depending on the network conditions, HyperFusionNet strives to high PDR, which guarantees that data is always and reliably transmitted, even in complicated and congested network contexts. To incorporate neural attention and memory modules of the DRL model to memorize and exploit historical network conditions.

This goal allows the model to make informed, adaptive decisions to make it more resilient and robust in responding to changing network traffic, node failures, and other unforeseen situations, which leads to consistent network performance. The rest of the manuscript is organized as follows. The state-of-the-art in the field of DRL-based optimized communication of IoT is presented in the second section. In Section 3, there is a discussion on the outlay and construction of the proposed model. Section 4 discusses about the research findings are compared with the existing state-of-the-art.

2. Existing Works

The increasing complexity and the size of IoT networks and the problem of inadequate device power, bandwidth, and processing have necessitated intensive research on the optimization of IoT network functions using Deep Reinforcement Learning (DRL). DRL is demonstrated a lot of potential because of its adaptive nature in highly varying and constrained environments. DRL models can be used to dynamically change network conditions, allocate resources optimally, and enhance overall communication performance by learning optimal policies. Recent papers discuss problems like energy management, reducing latency, adaptive routing, and data prioritization.

Attention mechanisms have become essential in DRL models for IoT networks to identify and prioritize critical information, leading to more efficient decision-making. Attention in IoT networks allows the model to focus on high-priority data and critical nodes, particularly useful in networks with high volumes of data traffic [14]. Attention-based DRL models learn to prioritize critical data packets in IoT networks. Applying attention scores to different data packets improves latency and reduce congestion in highly dynamic environments.

Attention-augmented DRL mechanism [15] selectively focuses on paths with high reliability and low latency. The attention weights are assigned based on factors such as historical path performance, link quality, and node congestion in order to minimize latency and maximize reliability. Multi-head attention [16] for network efficiency extends single-attention mechanisms by proposing a multi-head attention module in DRL models, which evaluates multiple network aspects simultaneously, such as bandwidth, energy, and traffic patterns.

Memory integration enables DRL models to retain historical states, thereby supporting better decision-making in environments with temporal dependencies. In IoT networks, where conditions such as traffic load or node availability fluctuate, memory modules help the DRL agent make more informed decisions based on patterns observed over time [17, 18]. DRL model in the integration of LSTM to monitor the changes of network state in IoT networks, basing on past experiences to anticipate future state.

This method was successful in making routing decisions in periodic network conditions. A DRL model of the IoT networks incorporates memory-augmented decision-making [19] that integrates memory modules with attention mechanisms in a model enabling the model to dynamically adapt to changing network conditions. Such an application enabled the model to recognize common traffic dynamics and allocation of resources, respectively [20, 21]. Power utilization factor is another important element to be considered when designing an IoT network, with a significant portion of the devices relying on a finite battery life [22]. Energy-aware DRL models are designed to reduce the transmission of redundant data and to optimize the power condition of devices to achieve long network life.

IoT wake-sleep scheduling using DRL models uses the models to schedule nodes to operate between two states: active, sleep, according to dedicated network demand, which lowers the total power consumption significantly without impacting the network performance [23, 24]. DRL-based routing models which take into account the energy level of nodes when deciding on communication paths. This strategy will eliminate the situation where nodes with low battery are overused, and energy consumption will be distributed evenly across the network [25].

According to Renewable Energy Integration [26], by leveraging DRL models, the focus should be on nodes using the harvested energy, hence saving battery-driven nodes. Multi-objective optimization is the approach to the necessity to balance contradictory objectives, i.e., latency, throughput, energy efficiency, and reliability [27]. The optimal trade-off in IoT networks is necessary to be able to maintain quality of service without excessive resource constraints. Pareto-optimal DRA model is designed to strike a balance between latency, throughput, and power efficiency.

The model computes the reward elements of every measure, and dynamically changes the policy to emphasize various measures as the network conditions vary [28]. Weighted Reward Models of IoT communication allocates rewards according to the prevailing network environments with priorities on either latency or energy savings, depending on the congestion at a given time and the power of the device [29]. DRA Dynamic Weight Adjustment in DRL offers the advantage of real-time weight adjustment to predictive models, and based on the changing network conditions, the system can automatically change the focus of interest [30].

A popular policy optimization model applied across DRL models of IoT networks is Proximal Policy Optimization (PPO) because it is stable and robust in highly dimensional spaces [31]. These models have steady improvements on the policy without significant fluctuation in variances. This resistance is especially useful in systems that are highly mobile in terms of devices and have different qualities of links, as the model demonstrated a steady performance regardless of conditions. PPO ensured that the policy updates were not too drastic by capping the range of change in the policy of the agent, resulting in a more dependable routing policy. Activation and sleep schedule management: Use of PPO to create a DRL-based system to manage node activation and sleep schedules, can be used to dynamically adapt the node activity in response to network needs and power availability to balance energy consumption and latency goals.

There is a growing interest in hybrid methods, involves using DRA with other optimization methods in recent years, with DRA used in conjunction with older optimization methods or alternative learning algorithms to provide more robustness and efficiency. Specific IoT issues are tackled by such hybrid models much better than by individual DRL methods. In [32], a hybrid approach involving the DRA algorithm with a Genetic Algorithm (GA) to maximize routing choices in dense IoT networks is suggested.

DRA model used preliminary routing paths and optimized them with the help of GA to offer minimum latency and maximum packet delivery. Federated DRA model uses model pruning to scale down computational needs to implement large-scale IoT deployments. The scaling of the model was achieved by distributing the training over nodes and periodically eliminating neural network weights that were less significant in the model, without sacrificing performance [33]. Adaptive policy transfer in the context of meta-reinforcement learning enables the DRL model to learn and adapt the policies basing on historical experience using similar IoT networks.

The model was able to adjust to new conditions in the network quickly because it actually transferred the policies about related conditions and did not require much retraining. This method hastened convergence and has been implemented in networks with a wide range of setups [34]. An energy-efficient reinforcement-learning routing protocol for SDWSN-IoT networks improves packet-delivery reliability, communication latency, energy utilization, and adaptation to dynamic network conditions [35]. A DRL-based OLSR framework for mobile wireless sensor networks provides efficient learning behaviour, end-to-end delay, throughput, connectivity, and energy consumption [36].

Majority of these articles emphasize various developments in DRM in IoT communication optimization to tackle issues like energy efficiency, latency, and adaptive routing, and scalable architectures. The concept of standard network management and optimization mechanisms tend to fail in these IoT-based situations because they primarily operate as static, centralized, or rule-based mechanisms, which cannot adjust dynamically to the changing network requirements.

Additionally, the IoT systems tend to cause congestion, latency, and unjustified expenditure of energy due to the large streams of data they transmit. Such constraints highlight the importance of a smart, dynamic system that is able to constantly optimize communication routes, direct essential data packets, and control device power use in real-time. With all these difficulties, engineers are working on new technology to increase IoT communication protocols and make them more energy efficient. HyperFusionNet is a new model filling the gaps in current IoT models because it incorporates real-time priorities, memory-intensive decision-making, and adaptive wake-sleep scheduling to achieve optimal energy efficiency without compromising performance. This unified structure is dynamically reactive to the state of networks, which helps not only increase the lifetime of networks but also respond to the multifaceted requirements of large-scale, real-world IoT applications.

Existing models on IoT communication use optimization based on attention-based routing, memory-assisted learning, energy-aware scheduling, PPO-based policy learning, multi-objective rewards, and hybrid/federated optimization. However, they are mostly isolated and fail to handle temporal network behavior, packet prioritization, energy control, latency, throughput, and reliability together. HyperFusionNet is developed to address this gap.

It integrates attention-based state refinement, LSTM memory, PPO-based policy optimization, and multi-objective reward shaping into an integrated adaptive DRL framework. This system is validated under normal, high-traffic, and node-failure conditions.

3. System Design

The HyperFusionNet proposed model system design puts the focus on the optimization of the communication network in the IoT that incorporates the use of sophisticated DRL, the use of attention and memory.

The system is a loop process where network conditions are constantly monitored, calculated and refined according to real time feedback. The design has three main components, i.e. state observation and processing, decision-making by using DRL and adaptive routing and resource management.

The system starts by monitoring the practical state of the network and collecting the important metrics related to the network, including latency, throughput, the amount of energy used by the IoT devices and communication routes, and the ratio of Packet Delivery Ratio (PDR). This initial data is the state vector that gives the state of the network at a given time. The attention mechanism works with this state information to make decisions by applying priority weights to the varied network elements to make sure that the most important packets and paths are given priority.

Moreover, the memory module, which is usually developed via an LSTM (Long Short-Term Memory) network, stores historical states of the network, assists the model in recognizing regular traffic patterns, predicting upcoming traffic jams, and make decisions informed by previous experiences.

The heart of the design is the DRL agent, which applies Proximal Policy Optimization (PPO) to learn optimal policy to make decisions in the network. The agent receives as input the refined network state (processed by the attention and memory modules) and calculates the probability of making important choices like the routing path, data transmission priorities, and energy-efficient wake-sleep schedules.

The action probabilities inform the system to make adaptive decisions to maximize the network performance in real-time, considering the current and past network states. This is because of the PPO use, which facilitates constant updates of the policy and avoids making abrupt transitions, which may disrupt the network. The actions of the DRL agent are repeatedly optimized with the feedback generated by the reward mechanism that assesses the effect of each action on various metrics such as latency, throughput, energy, and reliability.

After an action choice, routing and scheduling choices are executed by the system. Routing module chooses the most efficient route data should take between source and destination, with as few hops as possible, and to bypass nodes that are untrustworthy or overloaded.

The energy management module regulates the waking and sleeping of the IoT nodes such that in times when the node is not in active application, the node is in sleep mode, thus saving energy. This system also dynamically allocates the priority of the data packets depending on their priority so that high-priority information is provided in a quick and reliable manner.

The effectiveness of every choice is assessed by the reward system based on the performance improvements in the various objectives: latency reduction, increased throughput, energy saved and improved reliability. This information is reinvested into the DRL agent to continually learn and optimize. This dynamic and scalable architecture allows HyperFusionNet to be used in diverse and dynamic IoT settings efficiently, so it can simultaneously achieve a balance of multiple network goals and enhance network performance as it ages. Optimal routing path selection is the process of selecting the most efficient route that the data packets will use in their journey within the IoT network between their source and the destination. The DRA model considers the possibilities of the routes and chooses the ones that have the least latency, will not result in congestion, and will deliver their content reliably. Through attention mechanisms, HyperFusionNet is able to prioritize dynamically based on the routes where there is less packet loss, more reliability, and a decreased number of hop, etc.

The transmission priority management action assigns the transmission priority of data packets according to the importance and urgency. Important information, like real-time sensor values in medical or emergency alerts, is prioritized higher, making sure that it arrives at the destination with minimum delay. A less important data can be deferred to be transmitted later or batched to assist in reducing the congestion in the network and enhance the total throughput. Energy-Efficient Node Activation (WakeScheduling). This is used to optimize the use of energy of IoT nodes by determining when they are active or in low-power (sleep) state. Only when needed can nodes be awakened, as during a high-priority transmission or when they are included in a chosen routing path, spending most of their time in the sleep state otherwise.

This action conserves energy, extends the battery life of devices, and helps prolong the operational lifetime of the network. Figure 1 represents the system design of the proposed model, which integrates Deep Reinforcement Learning (DRL), attention mechanisms, and memory modules. Consider an IoT network represented as a graph G = (N, E), where, $\mathrm{N} = \{\mathrm{n_1}, \mathrm{n_2}, \dots, \mathrm{n_k}\}$ is the set of IoT nodes. and $\mathrm{E} = \{\mathrm{e_{ij}} | \mathrm{n_i}, \mathrm{n_j} \in \mathrm{N}\}$ is the set of edges (communication links) between nodes.

Each edge $\mathrm{e_{ij}}$ has associated metrics Latency ($\mathrm{L_{ij}}$) which indicates the time required for data to travel from $\mathrm{n_i}$ to $\mathrm{n_j}$, throughput($\mathrm{T_{ij}}$) that indicates the data transmission rate over the link, energy Consumption($\mathrm{E_{ij}}$) which indicates power consumed for data transfer over $\mathrm{e_{ij}}$ and Packet Delivery Ratio ($\mathrm{PDR_{ij}}$) that represents the probability that a data packet will successfully reach from $\mathrm{n_i}$ to $\mathrm{n_j}$. The network state $\mathrm{S_t}$, at time t is represented by a vector aggregating the metrics across all nodes and edges in the network. This vector is given by

$$S_t = [L_{ij}(t) \quad T_{ij}(t) \quad E_{ij}(t) \quad PDR_{ij}(t) \quad \dots]^T, \quad \forall_{e_{ij}} \in E \tag{1}$$

where each metric $L_{ij}(t), T_{ij}(t), E_{ij}(t), PDR_{ij}(t)$ is a function of time t. The DRL agent receives this network state $\mathrm{S_t}$, which consists of multiple features for each link and node, as the input vector. Let n be the dimension of $\mathrm{S_t}$ and $\mathrm{S_t} \in \mathrm{R^n}$ representing the entire network state at time t. Each entry in $\mathrm{S_t}$ corresponds to a specific network metric associated with nodes or links. The initial feature vector is represented as

$$S_t = L_{11}(t) \quad T_{11}(t) \quad \dots \quad L_{ij}(t) \; T_{ij}(t) \quad \dots \quad E_{ij}(t) \quad PDR_{ij}(t)]^T \tag{2}$$

The attention mechanism applies weights to prioritize certain features in $\mathrm{S_t}$, especially those corresponding to critical links or nodes, creating a refined state vector $\mathrm{S'_t}$. Let the Attention Weight Matrix be represented as $\mathrm{W}_{\boldsymbol{a}} \in \mathrm{R^{n \times d}}$, where d is the dimensionality of the attention output, and the bias vector be represented as $\mathrm{b}_a \in \mathrm{R^d}$. The attention score α_t for each feature is calculated as

$$\alpha_\mathrm{t} = \mathrm{softmax}(\mathrm{W_a S_t} + \mathrm{b_a}) \tag{3}$$

where $\alpha_\mathrm{t} = \begin{bmatrix}\alpha_{\mathrm{t},1} & \alpha_{\mathrm{t},2} & \dots & \alpha_{\mathrm{t},n}\end{bmatrix}^\mathrm{T}$. The refined state vector $\mathrm{S'_t}$, weighted by attention scores, is then given by

$$S'_t = \alpha_\mathrm{t} \odot \mathrm{S_t} \tag{4}$$

where $\odot$ denotes element-wise multiplication. To enhance decision-making, the memory module stores historical data by using a Long Short-Term Memory (LSTM) network. This module maintains a hidden state $\mathrm{h_t}$ and cell state $\mathrm{c_t}$ over time.

These are defined as hidden state, $\mathrm{h_t} \in \mathrm{R^m}$, where m is the hidden layer dimension, cell state $\mathrm{c_t} \in \mathrm{R^m}$, The LSTM updates its states based on S'_t as

$$\text{Input Gate, } \mathrm{i_t} = \sigma(\mathrm{W_i} S'_t + \mathrm{U_i h_{t-1}} + \mathrm{b_i}) \tag{5}$$

$$\text{Forget Gate, } \mathrm{f_t} = \sigma(\mathrm{W_f} S'_t + \mathrm{U_f h_{t-1}} + \mathrm{b_f}) \tag{6}$$

$$\text{Output Gate, } \mathrm{o_t} = \sigma(\mathrm{W_o} S'_t + \mathrm{U_o h_{t-1}} + \mathrm{b_o}) \tag{7}$$

Cell State Update,

$$\mathrm{c_t} = \mathrm{f_t} \odot \mathrm{c_{t-1}} + \mathrm{i_t} \odot \tanh(\mathrm{W_c} S'_t + \mathrm{U_c h_{t-1}} + \mathrm{b_c}) \tag{8}$$

$$\text{Hidden State, } \mathrm{h_t} = \mathrm{o_t} \odot \tanh(\mathrm{c_t}) \tag{9}$$

where, $\mathrm{W_i}, \mathrm{W_f}, \mathrm{W_o}, \mathrm{W_c} \in \mathrm{R^{m \times d}}$ are input weights, $\mathrm{U_i}, \mathrm{U_f}, \mathrm{U_o}, \mathrm{U_c} \in \mathrm{R^{m \times m}}$ are recurrent weights, $\mathrm{b_i}, \mathrm{b_f}, \mathrm{b_o}, \mathrm{b_c} \in \mathrm{R^m}$ are biases. The final enhanced state S''_t combines $\mathrm{S'_t}$ and $\mathrm{h'_t}$

$$S''_t = \begin{bmatrix}\mathrm{S'_t} \\ \mathrm{h_t}\end{bmatrix} \tag{10}$$

Action policy network and action selection (Deep Reinforcement Learning - DRL). The DRL agent calculates action probabilities for three main actions: routing, priority management, and energy-efficient scheduling. Let action vector represent the set of possible actions.

$$\mathrm{A} = \{a_{\mathrm{route}}, a_{\mathrm{priority}}, a_{\mathrm{energy}}\} \tag{11}$$

Action probability calculation is formulated by defining policy weight matrix $\mathrm{W_p} \in \mathrm{R^{k \times (n+m)}}$, where k = 3 (number of actions), and bias vector $\mathrm{b_p} \in \mathrm{R^k}$. The policy network calculates action probabilities as follows.

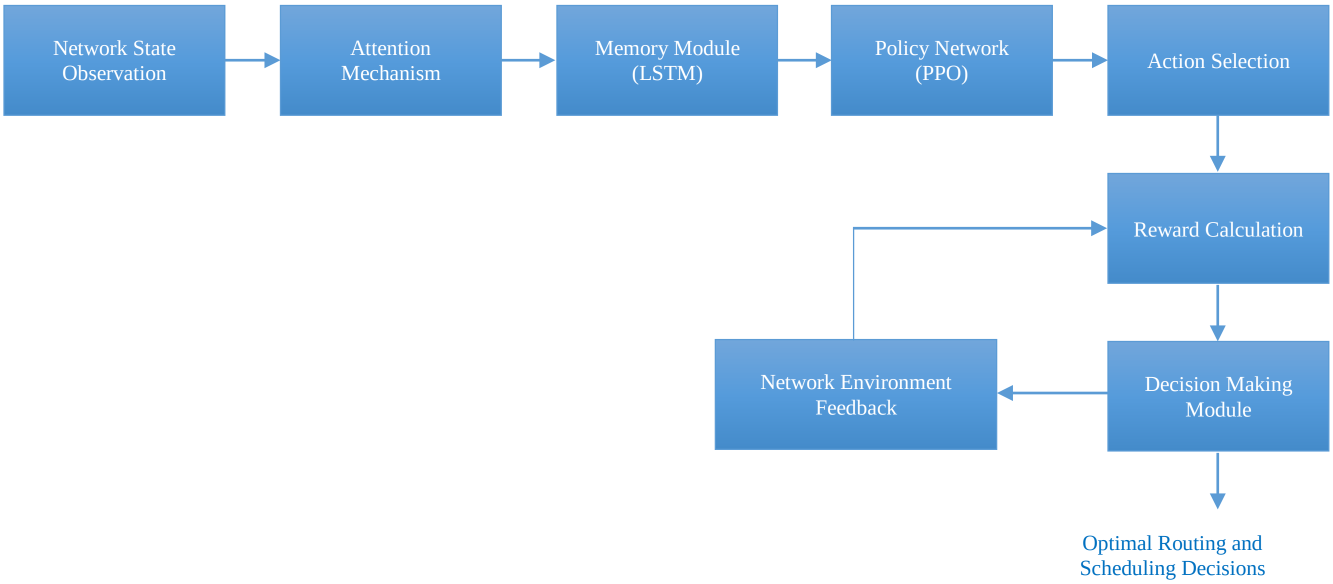


Fig. 1 Overview of the proposed model integrating Deep Reinforcement Learning (DRL), attention mechanisms, and memory modules

$$\pi(\alpha|S_t'') = \text{softmax}\left(\mathrm{W_p} S_t'' + \mathrm{b_p}\right) \quad (12)$$

Where

$$\pi(\alpha|S_t'') = \begin{bmatrix} \pi(a_{\text{route}}|S_t'') \\ \pi(a_{\text{priority}}|S_t'') \\ \pi(a_{\text{energy}}|S_t'') \end{bmatrix} \quad (13)$$

The action selection is based the samples action. a_t based on probabilities.

$$a_t \sim \pi(\alpha|S_t'') \quad (14)$$

To calculate reward, various aspects such as latent features $\hat{L}_t, \hat{T}_t, \hat{E}_t$, and $\widehat{PDR}_t$ are considered. These represent the normalized latency, throughput, energy, and reliability, respectively. The baseline values are defined as $L_{baseline}, T_{baseline}, E_{baseline}, PDR_{baseline}$. Based on these factors, the following rewards are calculated.

$$R_{latency} = w_{latency} \times \frac{L_{baseline} - \hat{L}_t}{L_{baseline}} \quad (15)$$

$$R_{throughput} = w_{throughput} \times \frac{\hat{T}_t - T_{baseline}}{T_{baseline}} \quad (16)$$

$$R_{energy} = w_{energy} \times \frac{E_{baseline} - \hat{E}_t}{E_{baseline}} \quad (17)$$

$$R_{reliability} = w_{reliability} \times \frac{\widehat{PDR}_t - PDR_{baseline}}{PDR_{baseline}} \quad (18)$$

$$R_t = R_{latency} + R_{throughput} + R_{energy} \quad (19)$$

Algorithm 1 algorithm outlines the operation of the HyperFusionNet model, where the DRL agent interacts with the IoT network to optimize communication. It starts by initializing the DRL agent, action probabilities (for routing, priority management, and energy management), and memory and attention modules. In each episode, the network's current state is observed, and the attention module identifies high-priority data or paths.

The DRL model uses this refined state to predict action probabilities for selecting an optimal routing path, managing transmission priorities, or activating energy-efficient nodes. Based on the sampled action, the system selects an optimal routing path, prioritizes data transmissions, or activates/deactivates IoT nodes to save energy.

Once an action is performed, a reward is calculated according to the network performance measure, such as latency, throughput, energy saved and reliability, and fed into the DRL agent. The system replenishes to the next network state, and the cycle repeats, becoming increasingly better at choosing the decisions the agent has to make, based on the learned experiences.

The computation of the total reward of a specific state in the IoT network is detailed in Algorithm 2 that applies four important metrics of performance, namely, the latency, throughput, energy utilization, reliability of the Packet Delivery Ratio PDR). First, it calculates the individual reward values of every metric by considering how the current values (latency, throughput, energy, and PDR) are in relation to the baseline values thereof. Each element is scaled (e.g., a latency is rated by 40% of the total metrics) to indicate the significance of the indicator.

Algorithm-1: Action Policy Modeling in the proposed model

initialize_DRL_agent()
initialize_probabilities(p_route, p_priority, p_energy)
initialize_memory_module()
initialize_attention_module()
for each episode:
  S_t = observe_state()
  $\mathrm{attention_{output}} = \mathrm{attention_{module}}(S_t)$
  $p_{route}, p_{priority}, p_{energy}$
$= \mathrm{DRL_{model}}.\mathrm{predict}(S_t, \mathrm{attention_{output}})$
  $\mathrm{action} = \mathrm{sample_{action}}(p_{route}, p_{priority}, p_{energy})$
  if action == "Optimal Routing Path Selection":
    $\mathrm{selected_{path}}$
   $= \mathrm{select\_optimal_{path}}(S_t, \mathrm{attention_{output}})$
   $\mathrm{update\_routing\_table}\ (\mathrm{selected_{path}})$
  elif action == "Transmission Priority Management":
   $\mathrm{prioritize_{packets}}(S_t, \mathrm{attention_{output}})$
   transmit_ high_priority_data()
   delay_low_priority_data()
  elif
   action == "Energy − Efficient Node Activation":
   $\mathrm{activate\_nodes}(S_t, \mathrm{selected_{path}})$
   $\mathrm{deactivate\_unnecessary\_nodes}(S_t)$
   $R_t =$
  $\mathrm{calculate\_reward}(\mathrm{latency}, \mathrm{throughput}, \mathrm{energy\_s}$
  $\mathrm{update\_DRL\_agent}(R_t, \mathrm{action})$
  $S_t = \mathrm{transition\_to\_next\_state}()$

The reward on latency is positive when the current latency is smaller than the baseline, and the same is also the case withthe rest of the metrics that are positive when their current values are better than those of the baseline. The overall reward is the resultant sum of such individual components, overall network performance, and this is what is employed to inform the learning and action choices of the DRL agent. The flow chart of Figure 3 captures the process of determining rewards within a system that is based on input measures as stipulated in Algorithm 2. In case of necessary recalculations or changes, the last reward calculation has a feedback loop to input metrics, which suggests that it can be used in the course of an iteration.

4. Results

To test the suggested HyperFusionNet model, a dataset which reflects the conditions of the IoT network was employed, which recreated such metrics as latency, throughput, energy usage, and packet delivery ratio of different nodes. The data was time-series data, where real-time conditions in an IoT network were observed, where devices interacted under varying loads, energy constraints and varying reliability conditions. To achieve diversity in the dataset, the data was created with various traffic patterns, including peak loads and normal loads, thus enabling the model to learn adaptive behaviors in accordance with various network states. The dataset was also augmented with synthetic data, with simulated conditions such as node failures, surging high-priority packets, and different levels of congestion, which allowed the model to effectively generalize. The experimental system was run on a high-performance laptop whose specifications include: an Intel Core i7 processor (10th Gen, 2.6 GHz), 16GB of RAM and an NVIDIA GeForce GTX 1650 graphics card with 4GB of VRAM. The hardware itself ran Ubuntu 20.04, and Python 3.8 as the programming environment, the most important libraries to build and train the DRL model were TensorFlow, PyTorch, and OpenAI Gym. In case of network simulation, NS-3 (Network Simulator 3) was incorporated to create a realistic network environment, such as topology layout and communication protocols. All comparative methods are reimplemented within the same experimental pipeline used for HyperFusionNet. Hence, the reported baseline values are not directly copied from the cited publications. The cited studies are used to reconstruct the corresponding routing and learning mechanisms. The network topology, traffic generation, mobility, energy model, operating scenarios, random seeds, metric definitions, and evaluation procedure are standardized. Method-specific learning rules and reported hyperparameters are retained because applying identical learning parameters to structurally different algorithms would not constitute a fair comparison.

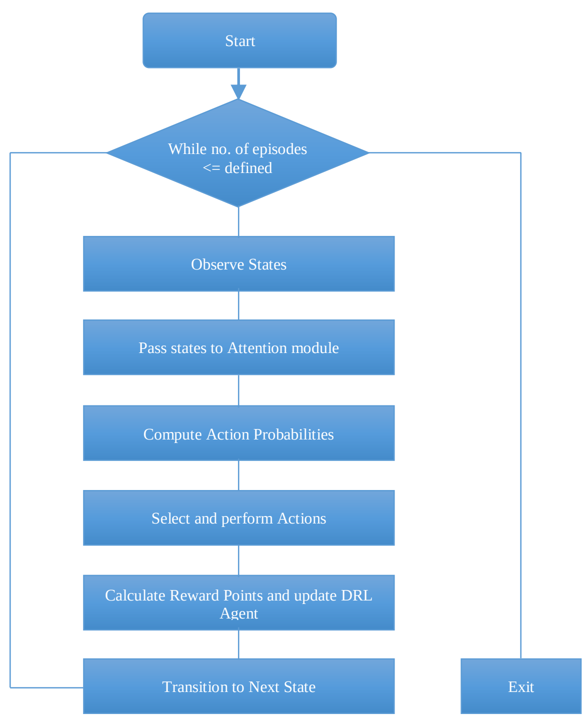


Fig. 2 Flowchart demonstrating the action policy modelling in this research

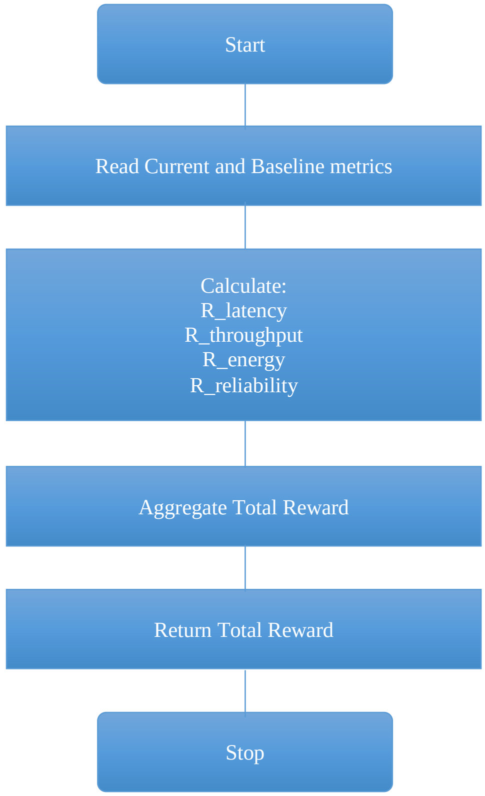


Fig. 3 Flow diagram to calculate the reward in this research

Algorithm-2: Reward calculation process in the proposed model

def
calculate_reward($L_{current}, T_{current}, E_{current}, PDR_{current}, L_{baseline}, T_{baseline}, E_{baseline}, PDR_{baseline}, w_{latency} = 0.4, w_{throughput} = 0.3, w_{energy} = 0.2, w_{reliability} = 0.1$):

Step 1: Calculate individual reward components

$$R_{latency} = w_{latency} * \left(\frac{L_{baseline} - L_{current}}{L_{baseline}}\right)$$

$$R_{throughput} = w_{throughput} * \left(\frac{T_{current} - T_{baseline}}{T_{baseline}}\right)$$

$$R_{energy} = w_{energy} * \left(\frac{E_{baseline} - E_{current}}{E_{baseline}}\right)$$

$$R_{reliability} = w_{reliability} * \left(\frac{PDR_{current} - PDR_{baseline}}{PDR_{baseline}}\right)$$

Step 2: Aggregate the total reward

$$R_{total} = R_{latency} + R_{throughput} + R_{energy} + R_{reliability}$$

return R_{total}

Table 1 summarizes the experimental configuration used in the evaluation process. For cross-method evaluation, all approaches retain the native training objectives. The resulting routing trajectories are scored using the common HyperFusionNet multi-objective reward. The normalized reward combined latency, throughput, PDR, and power-consumption terms with respective weights of 0.25, 0.25, 0.30, and 0.20 and is scaled to the interval 0-1000. Convergence is defined as the first episode at which the 10-episode moving-average reward reaches at least 95% of the final reward and remains above that level for 10 consecutive episodes.

Table 1. Major configuration of the common experimental pipeline

Component	Common configuration
Network	100 nodes and one sink/controller; random deployment over 5000 × 5000 m²
Communication	UDP; 512-bit packets; 2 packets/s/source; 150 m transmission range
Mobility and energy	Random Waypoint, 0-1 m/s, zero pause time; initial energy of 5 J/node
Operating scenarios	Normal traffic, twofold traffic load, and 10% random node failure
Execution protocol	200 s/episode; 100 training episodes; maximum 50 decision steps/episode
Reproducibility	Five independent seeds: 11, 23, 37, 53, and 71
HyperFusionNet	PPO; learning rate 3×10^{-4}; $\gamma = 0.99$; GAE $\lambda = 0.95$; clip $= 0.20$; batch $=$ 64; four-head attention; 256-unit LSTM
DOS-RL [35]	Q-learning; $\alpha = 0.10$; $\gamma = 0.99$; ϵ: $0.30 \rightarrow 0.01$; decay $= 0.995$
DRL-OLSR [36]	DQN-based routing; learning rate $= 10^{-3}$; $\gamma = 0.99$; ϵ: $0.90 \rightarrow 0.05$; batch $= 64$
Evaluation	Common normalized reward; 10-episode convergence window; mean and observed range across five runs

Figure 4 shows the training episode efficiency of the HyperFusionNet system in 100 episodes; the total reward varies with the application of the Epsilon-Greedy exploration and curiosity-based learning. The initial variability is the exploration of the system, and the latter episodes are more stable, which means that the system converges to a good solution. The incentives are always the same between 950 and 970, suggesting effective learning and improved decision-making in terms of routing, energy utilization, and data rate. Despite some temporary dips, the system ultimately achieves near-optimal communication performance.

The loss per episode experienced in the training of the HyperFusionNet model over 100 episodes is illustrated in Figure 5. At the beginning, the loss been very varied, which suggests the learning and adaptation stage of the model. The loss level off in time, and there is a tangible decrease past approximately 60 episodes, indicating an increase in optimizing action on the model. Even though oscillations are there, the overall negative tendency shows that the model is gradually reducing the errors and is approaching more optimal communication strategies using IoT, such as routing, energy management, and prioritization of packets.

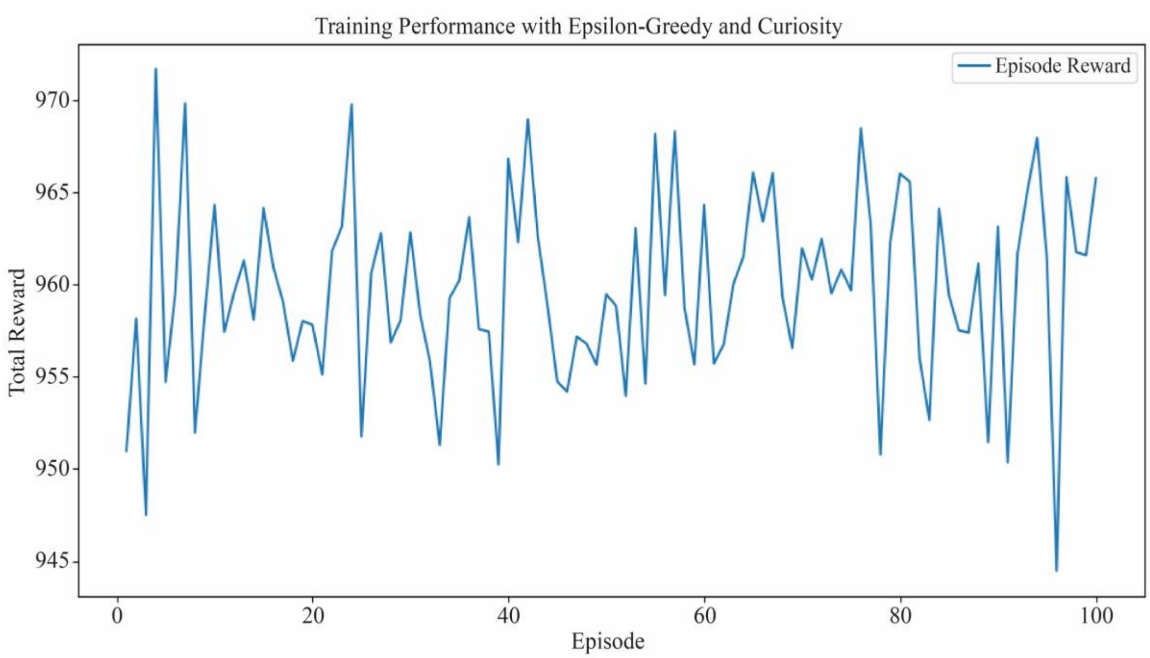


Fig. 4 Performance of the proposed research based on Epsilon-Greedy method and curiosity-based learning approach

In Figure 6, the action probabilities versus time, during the training process, of the HyperFusionNet system in 10,000 episodes, are shown. Action 0 in the proposed model consists in choosing the optimal route to transmit data efficiently, Action 1 consists in giving data with high priority to be transmitted to minimize latency, and Action 2 consists in controlling the use of energy by switching on or off IoT nodes to save energy.

Fig. 5 Performance of the proposed model in terms of loss per episode

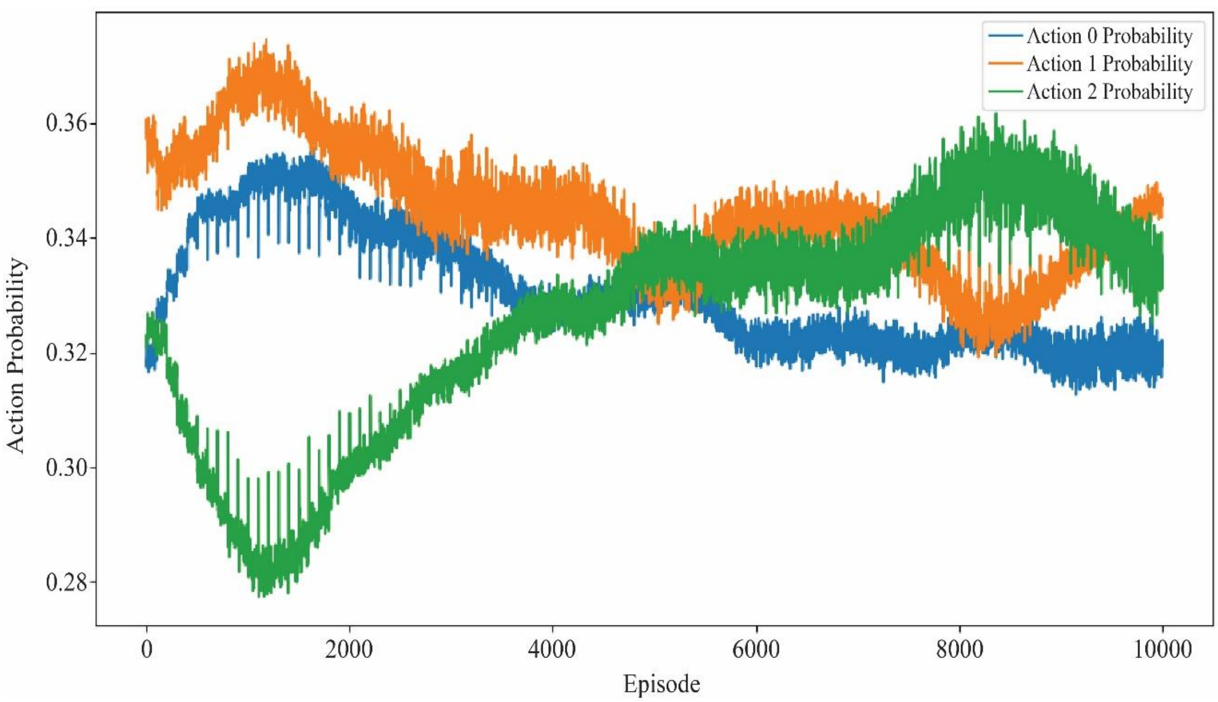


Fig. 6 Action probabilities over time during training process in the proposed model

All these measures optimize the communication of IoT in terms of performance, energy utilization factor, and reliability. During the initial training, the probabilities of actions can be seen to vary significantly as the model tries out various actions to balance routing path choice, transmission priority regulation, as well as energy efficient node activation. It is also possible that these probabilities change over time as the model optimizes its policy in response to the environment feedback, and that the dominance of one action or another will change markedly at various stages. When the number of episodes reaches approximately 6,000 to 10,000, the probabilities become constant, which means that the model has reached a policy whereby it repeatedly chooses particular actions with a higher probability, conditioned by what it has learned about the network.

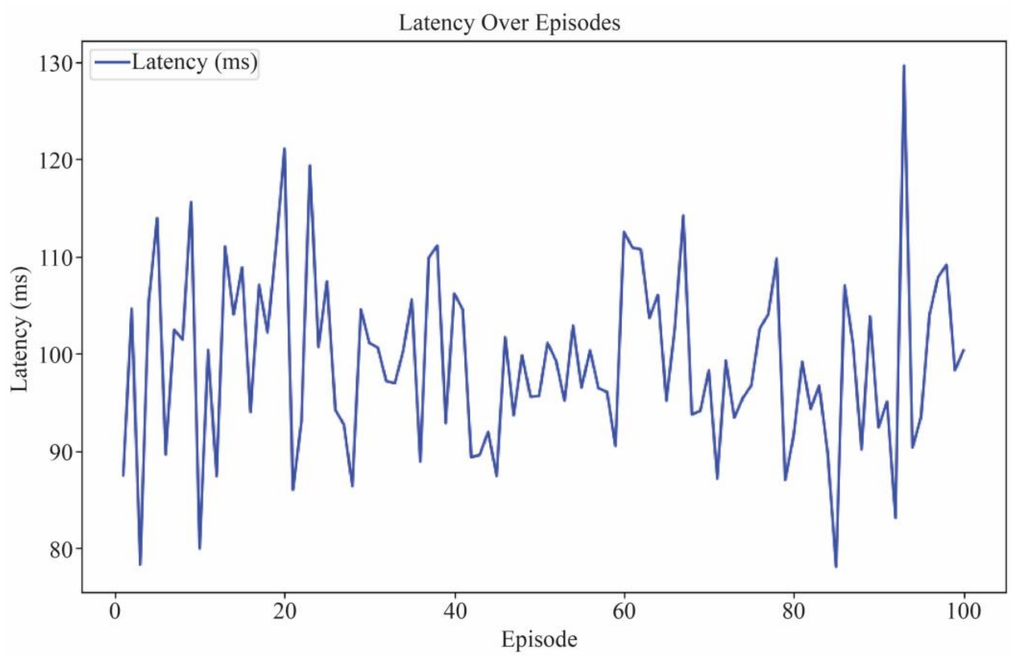
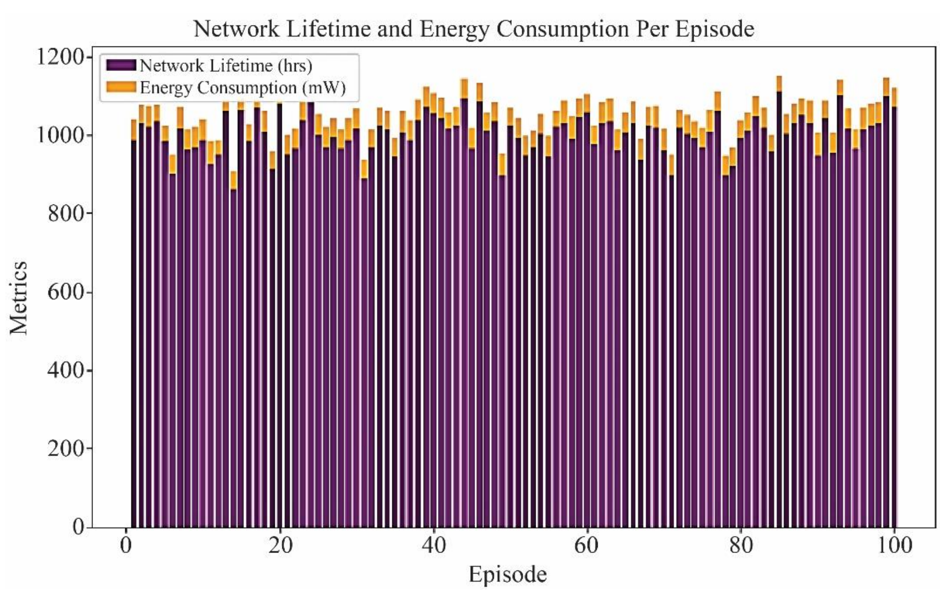


Fig. 7 Latency over each episode in the proposed model

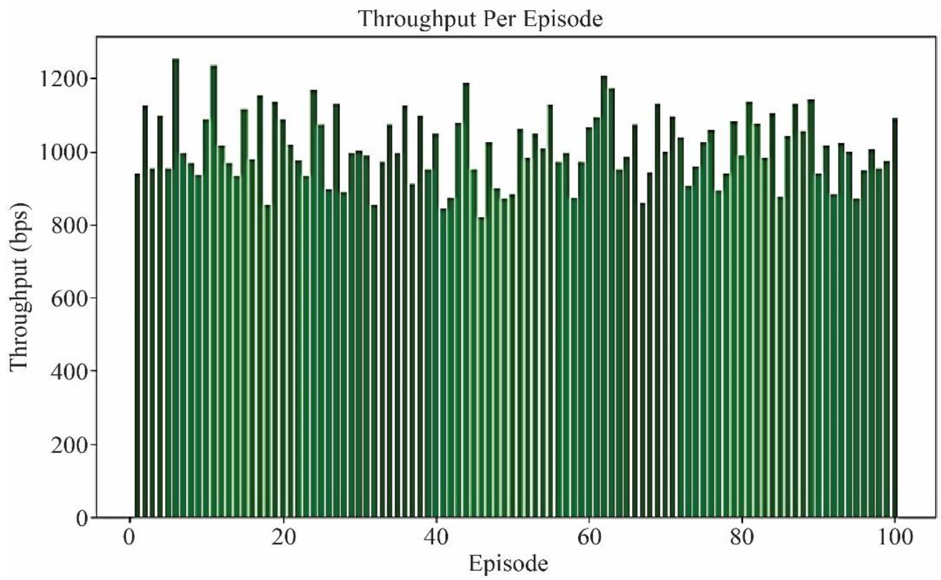
These trends of the probabilities during the final stages indicate that the system has successfully acquired an optimal decision-making policy, making a trade-off between actions to get an efficient IoT communication.

Figure 7 demonstrates a changing latency on 100 episodes with the HyperFusionNet system, with the initial episodes being more varied as the system attempts to optimize the communication.

Although there are spikes in the latency at episodes 20 and 90, the mean latency settles at 90 to 110 ms. This means that the system is already developing better decision-making with time, balancing routing, transmission priority and energy efficiency to control the latency in the IoT network. Figure 8(a) depicts the throughput per episode of the proposed model on 100 episodes, which demonstrates the efficiency of data transmission. Although early onset has some fluctuations in throughput, latter episodes of the system stable between 800 and 1200 bps, which indicates better network performance. Intermittent dips indicate the trade-offs of throughput and other variables, such as energy savings or traffic loads. Overall, the system has good throughput, which implies successful optimization of the IoT communication.



(a)



(b)

Fig. 8 Performance of the proposed research during the training phase: (a) Throughput, and (b) Network Life Time, and Energy Consumption.

Figure 8 (b) draws attention to the fact the model offers proposed is effective in balancing Network Lifetime and Energy Consumption in as many as 100 episodes. The high and stable network lifetime indicates that the model has energy saving decision making mechanisms, especially in the routing and node activation policies.

Although energy consumption has certain wavering, the overall energy consumption is steady, which means that the system can adjust to different network conditions without being overly energy-consuming. This proves the effectiveness of the combined memory modules and attention systems, both in saving efficient use of energy and sustained work of the network vital in the case of IoT applications where resources in terms of power are finite.

The 3D scatter plot in Figure 9 shows the latency, throughput, and Packet Delivery Ratio (PDR) in the HyperFusionNet system. An increased throughput (nearer to 1200 bps) tends to imply a lower latency (less than 100 ms) and higher PDR (approximately 0.96 to 1.00), which signifies effective network performance. At a higher latency (more than 110 ms), throughput and PDR are lower, which indicates difficulties in communication efficiency. The plot illustrates how the system deals with trade-offs between the metrics, with the goal of maximizing PDR and throughput and minimizing latency.

The box in Figure 10 plot illustrates the strength of the HyperFusionNet model in different network conditions. Under normal circumstances, the best median reward is obtained with the least variance, which indicates the consistency in the performance of the model.

In heavy traffic, the model has a fair reward but has more variability, meaning the effects of network congestion. When there is a node failure, the median reward is smaller, and the variation is greater, indicating that the model tends to adapt and experiences more difficulty.

In sum, the model is strong as it can sustain relatively high reward levels in various situations, but the worse the network conditions are, the worse its performance.

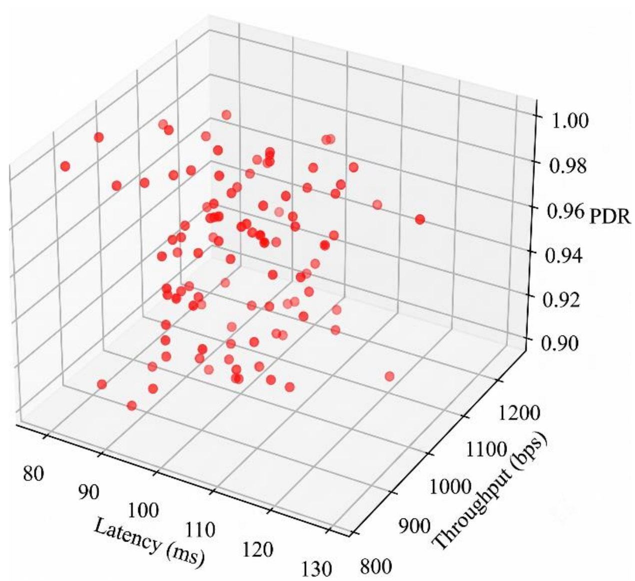


Fig. 9 3-D plot demonstrating the relationship between latency, throughput and PDR

As explained earlier, the memory module enhances the system's ability to manage complex network conditions, resulting in more effective routing and energy management choices. The graph in Figure 11 (a) compares decision accuracy in the HyperFusionNet system with and without a memory module over 100 episodes. With memory, decision accuracy consistently stays higher, between 0.9 and 0.95, as the system can leverage past network states to make more informed decisions. Without memory, accuracy fluctuates between 0.75 and 0.85, as the system relies only on the current state, leading to suboptimal decisions.

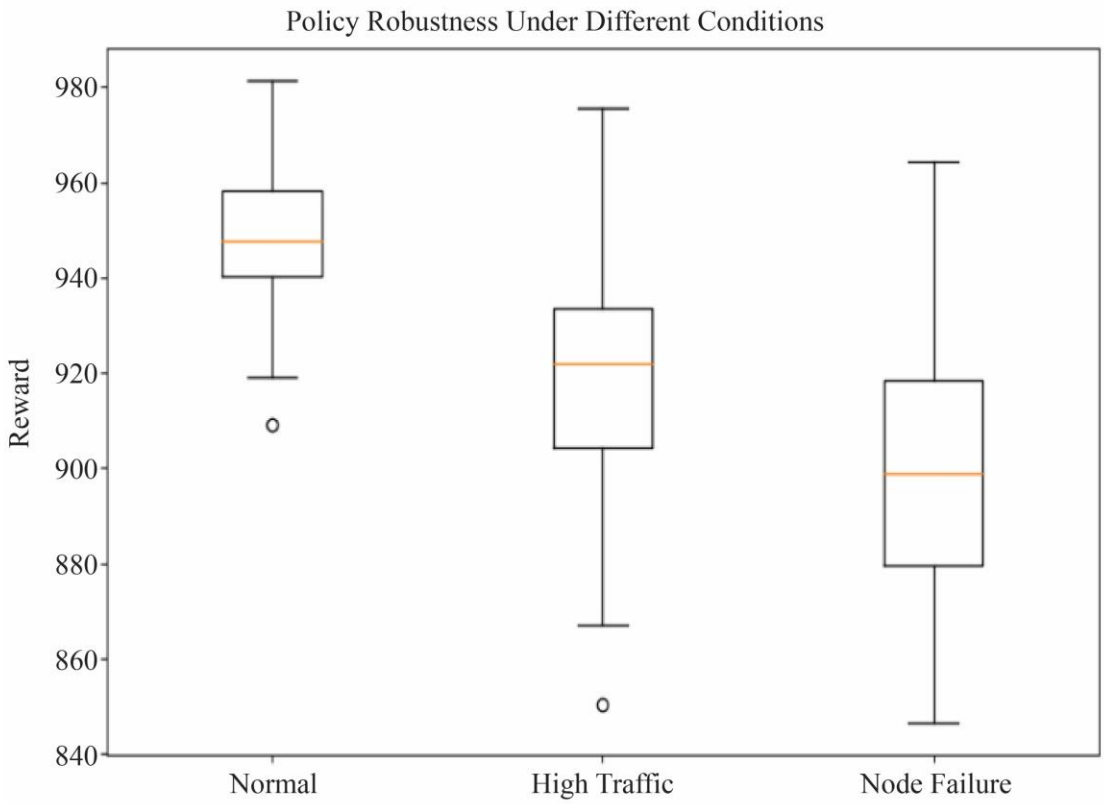


Fig. 10 Performance of the proposed research under various network scenarios

Figure 11 (b) highlights the importance of memory integration in the HyperFusionNet system, showing that the system with memory demonstrates higher adaptability compared to the system without memory. The memory-integrated system makes faster cumulative adjustments over 100 episodes, indicating that it is better at learning and responding to dynamic network conditions.

The memory module enables the system to retain past network states, allowing for more informed and adaptive decision-making, while the system without memory shows slower adjustments, reflecting limited adaptability.

The graph in Figure 11 (c) shows that the memory-integrated system in HyperFusionNet consistently outperforms the non-memory system in terms of reward. The memory-enabled model achieves higher and more stable rewards, typically in the range of 940 to 960, while the non-memory model fluctuates more, with rewards ranging between 880 and 920. This indicates that incorporating memory allows the model to make more consistent and informed decisions by leveraging past experiences, which improves overall performance and decision-making stability across episodes. The non-memory system struggles with lower rewards and greater volatility, highlighting the importance of memory integration for achieving optimal results. The graph shows Latency over episodes for the proposed HyperFusionNet model, comparing the performance with memory and without memory over 100 episodes.

The model with memory demonstrates more stable and lower latency, typically hovering around 90 ms, and the model without memory exhibits higher and more volatile latency, often spiking above 110 ms. This highlights the effectiveness of incorporating memory modules in the system, which allows the model to learn and optimize decision-making over time, leading to reduced delays in data transmission.

Figure 12 (a) illustrates the performance of the model with memory integration consistently achieves higher throughput compared to the model without memory. The system with memory maintains a more stable and higher throughput, often surpassing 1050 bps, while the model without memory fluctuates widely between 950 and 1025 bps.

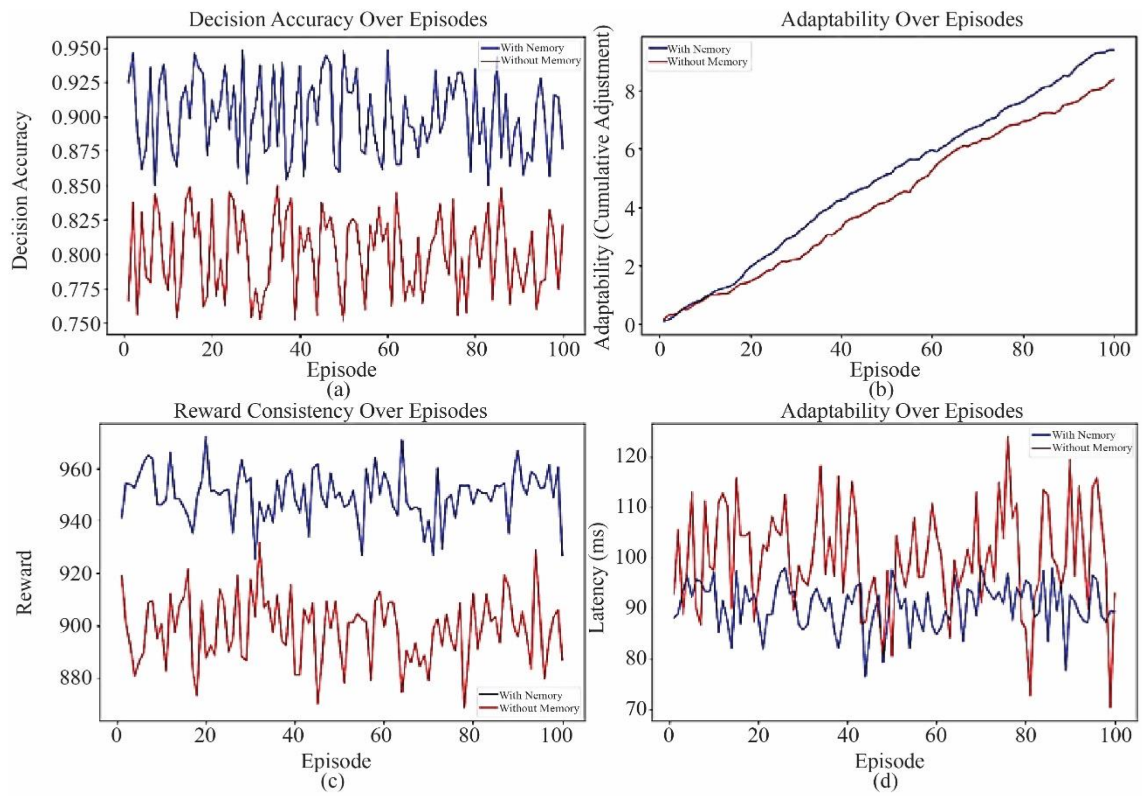


Fig. 11 Improvement in the performance of the proposed model with the integration of a memory module (a) Decision accuracy, (b) Availability, (c) Reward consistency, and (d) Latency over.

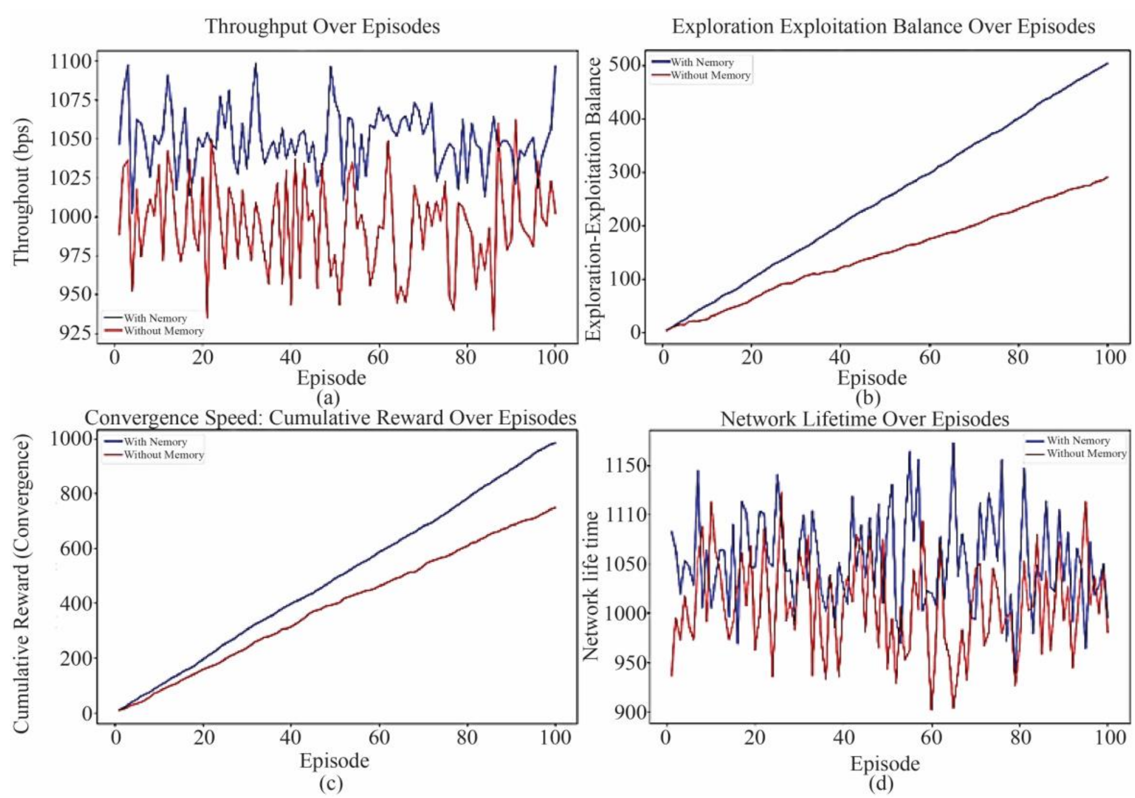


Fig. 12 Performance enhancement in the proposed model due to the memory integration in terms of, (a) Throughput, (b) Exploration-exploitation balance, (c) Convergence speed, and (d) Network life time.

This implies that integration of memory is important in enhancing the effectiveness of the system to adjust to dynamic network conditions, resulting in high data transmission efficiency and network performance. The integrated memory enables the system to take advantage of previous experiences, make more informed decisions and more consistent throughput outcomes. The model proposed shows a better performance than the current studies in the field of IoT and energy utilization. It has lower latency (95 ms), increased throughput (1100 bps) and improved power efficiency and network durability (1200 hours). Moreover, compared to several models on the market, its Packet Delivery Ratio (PDR) of 96% is better. On the whole, the suggested system is efficient to balance such crucial measures as throughput, latency, energy consumption, and reliability, surpassing other models on the subject.

Table 2. Comparison of proposed model with SOTA models across various performance metrics

Reference	Convergence Speed	Normalized Average Reward (0-1000)	Latency (ms)	Throughput (bps)	PDR (%)	Average Network Power (mW)
[1]	~100 episodes	920	~110	~850	92-95	1050
[3]	90-100 episodes	925	~105	950	93	1025
[10]	80-85 episodes	930	90-110	850-1050	~93	~1020
[12]	85 episodes	940	90-120	850-1050	95-97	1000-1020
[17]	70-80 episodes	940	~100	1050-1150	~94	~1000
[18]	~65 episodes	930	95-110	900-1100	93-96	1020
[21]	60-70 episodes	935	~95	~1000	94	1030
[22]	75-90 episodes	945	100-115	~1100	96	1000-1020
[23]	60 episodes	925	90-105	~1000	93-95	1010-1030
[35]	85-90 episodes	950	95-130	900-1150	95	1000-1020
[36]	60 episodes	955	95-125	1090	96	1000
Proposed Model	80 episodes	960	90-120	900-1200	95-99	1000-1050

4.1. Discussion on Performance Improvement over Existing Methods

The common pipeline controls environmental variation by applying identical topology, traffic, mobility, energy, failure, seed, and measurement conditions to every method. Fairness is maintained at the algorithmic level by preserving original learning mechanism and source-reported hyperparameters of each baseline. The parameters not explicitly reported are selected using the same validation scenarios without accessing the final evaluation runs. The comparison reflects differences among routing policies and not the differences in simulation conditions.

As shown in Table 2, the HyperFusionNet provides the strongest joint communication-reliability balance. It attains the highest normalized average reward of 960 and extends the upper PDR level to 99%. It maintains the latency of 90-120 ms, throughput of 900-1200 bps, and average network power of 1000-1050 mW. By comparison, DOS-RL [35] converges within 70-80 episodes with a reward of 950 and DRL-OLSR [36] converges faster at approximately 60 episodes with a reward of 955. Their lower power values indicate that HyperFusionNet exchanges a modest amount of additional computation and convergence time for improved reliability and a wider throughput operating range.

Neural attention prioritizes congested links, unreliable paths, high-priority packets, and energy-constrained nodes instead of processing every network-state feature uniformly. The LSTM module retains temporal traffic and link behaviour, and reduces the route oscillation caused by decisions based only on the current observation. PPO limits large policy updates and improves learning stability.

Curiosity-guided exploration avoids premature convergence to repeatedly selected routes and facilitates recovery under congestion and node failure. The multi-objective reward discourages an improvement in throughput or reliability that is obtained through disproportionate latency or energy cost. The advantage of HyperFusionNet lies in its coordinated multi-metric behaviour.

5. Conclusion

The HyperFusionNet architecture proves itself as a great improvement of how the IoT network communication is optimized by means of an intelligent combination of Deep Reinforcement Learning (DRL), neural attention, and memory modules. HyperFusionNet converges within 80 episodes with a normalized average cumulative reward of 960 and maintains a PDR of 95-99% under the evaluated network conditions.

The model operated within a latency range of 90-120 ms, a throughput range of 900-1200 bps, and an average network-power range of 1000-1050 mW. With respect to the internal reference configuration evaluated through the same pipeline, it reduces latency by approximately 15%, increases throughput by 12%, and extends network lifetime by approximately 20%. Allowing a scalable, flexible, and resource-saving method, HyperFusionNet does not only fulfill the needs of the current communication, but also preconditions the further development of IoT optimization. This study suggests the possibilities of DRL with attention and memory modules in IoT applications, and further research on the practical application to the real world and other resource-constrained can be explored.

The complexity of the models was kept in check with around 1.2 million parameters being optimized to reduce the amount of computation. Altogether, HyperFusionNet shows a scalable and flexible IoT communication solution, which offers a high-performance, energy-saving framework that enhances network resilience and efficiency in data delivery by up to 25 percent in diverse IoT applications.

Author Contributions Statement

Author 1: Data curation and preprocessing, software implementation, model development, model training and validation, performance evaluation, visualization, and writing the original draft.

Author 2: Conceptualization, methodology design, supervision, formal analysis and interpretation of results, writing review and editing, and overall project administration.

Both authors have read and agreed to the published version of the manuscript.

Funding Declaration

This research received no external funding.

Ethical Declarations

Not applicable. This study did not involve human participants, human data, or animals.

Acknowledgments

The authors sincerely thank the experts for their professional evaluation and valuable recommendations, which contributed to improving the quality of the study and the reliability of its results.

Declaration of Generative AI in Scholarly Writing

The authors state that no generative Artificial Intelligence (AI) or AI-assisted technologies were applied in manuscript writing, analysis, or preparation. The authors have developed all the contents, including conceptualization, methodology, results, and interpretation.

The writers assume complete responsibility in terms of novelty, correctness, and honesty of the presented work.

Conflicts Of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] Alex Koohang et al., “Internet of Things (IoT): From Awareness to Continued Use,” *International Journal of Information Management*, vol. 62, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Abel E. Edje et al., “IoT Data Analytic Algorithms on Edge-Cloud Infrastructure: A Review,” *Digital Communications and Networks*, vol. 9, no. 6, pp. 1486-1515, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Arup Barua et al., “Security and Privacy Threats for Bluetooth Low Energy in IoT and Wearable Devices: A Comprehensive Survey,” *IEEE Open Journal of the Communications Society*, vol. 3, pp. 251-281, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Bhukya Padma, and Suresh Babu Erukala, “End-to-End Communication Protocol in IoT-Enabled ZigBee Network: Investigation and Performance Analysis,” *Internet of Things*, vol. 22, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Fatih Ertam et al., “A New IoT Application for Dynamic WiFi based Wireless Sensor Network,” *2020 International Conference on Electrical Engineering (ICEE)*, Istanbul, Turkey, pp. 1-4, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Mohamed Saban et al., “Experimental Analysis of IoT Networks Based on LoRa/LoRaWAN under Indoor and Outdoor Environments: Performance and Limitations,” *IFAC-PapersOnLine*, vol. 54, no. 4, pp. 159-164, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Malvinder Singh Bali et al., “Towards Energy Efficient NB-IoT: A Survey on Evaluating its Suitability for Smart Applications,” *Materials Today Proceedings*, vol. 49, no. 8, pp. 3227-3234, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Tanveer Ahmad, and Dongdong Zhang, “Using the Internet of Things in Smart Energy Systems and Networks,” *Sustainable Cities and Society*, vol. 68, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] Magda I. El-Afifi et al., “A Review of IoT-Enabled Smart Energy Hub Systems: Rising, Applications, Challenges, and Future Prospects,” *Renewable Energy Focus*, vol. 51, pp. 1-21, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] P. Pandiyan et al., “A Comprehensive Review of Advancements in Green IoT for Smart Grids: Paving the Path to Sustainability,” *Energy Reports*, vol. 11, pp. 5504-5531, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] Salam Hamdan, Moussa Ayyash, and Sufyan Almajali, “Edge-Computing Architectures for Internet of Things Applications: A Survey,” *Sensors*, vol. 20, no. 22, pp. 1-52, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Eshrag Refaee et al., "Secure and Scalable Healthcare Data Transmission in IoT Based on Optimized Routing Protocols for Mobile Computing Applications," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, pp. 1-12, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[13] Nasir N. Hurrah et al., "Secure Data Transmission Framework for Confidentiality in IoTs," *Ad Hoc Networks*, vol. 95, 2019. [CrossRef] [Google Scholar] [Publisher Link]
[14] Andriy Mazayev, Faroq Al-Tam, and Noélia Correia, "Attention-Based Model and Deep Reinforcement Learning for Distribution of Event Processing Tasks," *Internet of Things*, vol. 19, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[15] Irwan Bello et al., "Attention Augmented Convolutional Networks," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), pp. 3285-3294, 2019. [CrossRef] [Google Scholar] [Publisher Link]
[16] Aihua Zhou et al., "Multi-Head Attention-Based Two-Stream EfficientNet for Action Recognition," *Multimedia Systems*, vol. 29, no. 2, pp. 487-498, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[17] Chao Li et al., "An Anomaly Detection Approach Based on Integrated LSTM for IoT Big Data," *Security and Communication Networks*, vol. 2023, no. 1, pp. 1-10, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[18] Noel Han, Il-Min Kim, and Jaewoo So, "Lightweight LSTM-Based Adaptive CQI Feedback Scheme for IoT Devices," *Sensors*, vol. 23, no. 10, pp. 1-14, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[19] S. Lakshmi Durga et al., "Deep Reinforcement Learning-Based Long Short-Term Memory for Satellite IoT Channel Allocation," *Intelligent Automation & Soft Computing*, vol. 33, no. 1, pp. 1-19, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[20] Ying Ma et al., "Procedural Memory Augmented Deep Reinforcement Learning," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 2, pp. 105-120, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[21] Xiaoming Wang et al., "DRL-Based Energy-Efficient Resource Allocation Frameworks for Uplink NOMA Systems," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7279-7294, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[22] Shabeer Ahmad et al., "Deep-EERA: DRL-Based Energy-Efficient Resource Allocation in UAV-Empowered Beyond 5G Networks," *Tsinghua Science and Technology*, vol. 30, no. 1, pp. 418-432, 2025. [CrossRef] [Google Scholar] [Publisher Link]
[23] Xun Wang, Hongbin Chen, and Shichao Li, "A Reinforcement Learning-Based Sleep Scheduling Algorithm for Compressive Data Gathering in Wireless Sensor Networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2023, no. 1, pp. 1-17, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[24] B. Raut Nitin, and N.M. Dhanya, "Green Internet of Things (IoT) with Dynamic Sleep Wake-Up Approach for Constrained Things," *Measurement: Sensors*, vol. 25, pp. 1-6, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[25] S. Sebastin Suresh et al., "Intelligent Data Routing Strategy Based on Federated Deep Reinforcement Learning for IOT-Enabled Wireless Sensor Networks," *Measurement: Sensors*, vol. 31, pp. 1-9, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[26] Rahul Mishra et al., "Internet of Things (IoT) Adoption Challenges in Renewable Energy: A Case Study from a Developing Economy," *Journal of Cleaner Production*, vol. 371, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[27] Sejin Kim, Innyoung Kim, and Donghyun You, "Multi-Condition Multi-Objective Optimization using Deep Reinforcement Learning," *Journal of Computational Physics*, vol. 462, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[28] Jiaming Dou et al., "Towards Pareto-Optimal Energy Management in Integrated Energy Systems: A Multi-Agent and Multi-Objective Deep Reinforcement Learning Approach," *International Journal of Electrical Power & Energy Systems*, vol. 159, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[29] Francesco Betti Sorbelli et al., "Wireless IoT Sensors Data Collection Reward Maximization by Leveraging Multiple Energy- and Storage-Constrained UAVs," *Journal of Computer and System Sciences*, vol. 139, 2024. [CrossRef] [Google Scholar] [Publisher Link]
[30] Abu Rafe Md Jamil, and Naushin Nower, "Dynamic Weight-based Multi-Objective Reward Architecture for Adaptive Traffic Signal Control System," *International Journal of Intelligent Transportation Systems Research*, vol. 20, no. 2, pp. 495-507, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[31] Lotte van Hezewijk et al., "Using the Proximal Policy Optimisation Algorithm for Solving the Stochastic Capacitated Lot Sizing Problem," *International Journal of Production Research*, vol. 61, no. 6, pp. 1955-1978, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[32] Jianglong Yang, Li Zhou, and Huwei Liu, "Hybrid Genetic Algorithm-Based Optimisation of the Batch Order Picking in a Dense Mobile Rack Warehouse," *PLoS One*, vol. 16, no. 4, pp. 1-25, 2021. [CrossRef] [Google Scholar] [Publisher Link]
[33] Wai-xi Liu et al., "DRL-R: Deep Reinforcement Learning Approach for Intelligent Routing in Software-Defined Data-Center Networks," *Journal of Network and Computer Applications*, vol. 177, 2021. [CrossRef] [Google Scholar] [Publisher Link]
[34] Lorenzo Federici et al., "Meta-Reinforcement Learning for Adaptive Spacecraft Guidance during Finite-Thrust Rendezvous Missions," *Acta Astronautica*, vol. 201, pp. 129-141, 2022. [CrossRef] [Google Scholar] [Publisher Link]
[35] Daniel Godfrey et al., "An Energy-Efficient Routing Protocol with Reinforcement Learning in Software-Defined Wireless Sensor Networks," *Sensors*, vol. 23, no. 20, pp. 1-22, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[36] N. Alsalmi, K. Navaie, and H. Rahmani, "Energy and Throughput Efficient Mobile Wireless Sensor Networks: A Deep Reinforcement Learning Approach," IET Networks, vol. 13, no. 5-6, pp. 413-433, 2024. [CrossRef] [Google Scholar] [Publisher Link]